

## CLAIMS

What is claimed is:

- 1) A method for selecting a value in a distributed computing system, the method comprising: receiving a first proposed value from a first client having a first client identifier; voting for the first proposed value in a first system step; transmitting a first indication of the voting for the first proposed value to one or more devices; and transmitting a first result of the voting for the first proposed value to the first client, wherein the voting for the first proposed value, the transmitting the first indication of the voting for the first proposed value, and the transmitting the first result are not performed if a second proposed value was proposed by a second client having a second client identifier that is more dominant than the first client identifier and the second proposed value was previously voted for.
- 2) The method of claim 1, wherein the first proposed value comprises a first function, and wherein the voting for the first proposed value comprises provisionally executing the first function in the first system step.
- 3) The method of claim 1, wherein the voting for the first proposed value comprises changing a previous vote for the second proposed value if the second proposed value was previously voted for and if the second client identifier is less dominant than the first client identifier.

4) The method of claim 1, wherein the first proposed value comprises a first function identified by a first function identifier, and wherein the voting for the first proposed value comprises executing the first function in the first system step unless the first function identifier is equivalent to a second function identifier that identifies a second function, wherein the second function was executed in a second system step that preceded the first system step.

5) The method of claim 1, wherein the first proposed value comprises a first idempotent function, and wherein the voting for the first proposed value comprises executing the first idempotent function in the first system step even if the first idempotent function is equivalent to a second idempotent function that was executed in a second system step that preceded the first system step.

6) The method of claim 1 further comprising: receiving a message, wherein the message is part of a fault tolerant consensus algorithm; ignoring additional proposed values from the first client; and participating in the fault tolerant consensus algorithm.

7) The method of claim 6, wherein the participating in the fault tolerant consensus algorithm comprises transmitting a possibly selected proposed value if a proposed value was previously voted for, wherein the possibly selected proposed value was previously voted for and was proposed by a client having a most dominant client identifier among all clients whose proposals were received and who proposed values for a current system step.

8) The method of claim 1 further comprising: transmitting one or more polling messages to initiate a fault tolerant consensus algorithm; receiving one or more vote indication messages in response to the one or more polling messages; and selecting, as a third proposed value, any value if the one or more vote indication messages indicate that at least one device has not previously voted or if the one or more vote indication messages indicate two or more different possibly selected proposed values, or selecting, as the third proposed value, a common possibly selected proposed value if all possibly selected proposed values indicated by the one or more vote indication messages are equivalent to the common possibly selected proposed value, wherein a possibly selected proposed value was previously voted for by a device and was proposed by a client having a most dominant client identifier among all clients whose proposals were received by the device, and wherein further the third proposed value is proposed using the fault tolerant consensus algorithm.

9) A computer-readable medium having computer-executable instructions for selecting a value in a distributed computing system, the computer-executable instructions performing steps comprising: receiving a first proposed value from a first client having a first client identifier; voting for the first proposed value in a first system step; transmitting a first indication of the voting for the first proposed value to one or more devices; and transmitting a first result of the voting for the first proposed value to the first client, wherein the voting for the first proposed value, the transmitting the first indication of the voting for the first proposed value, and the transmitting the first result are not performed if a second proposed value was proposed by a second client having a second client

identifier that is more dominant than the first client identifier and the second proposed value was previously voted for.

10) The computer-readable medium of claim 9, wherein the first proposed value comprises a first function, and wherein the voting for the first proposed value comprises provisionally executing the first function in the first system step.

11) The computer-readable medium of claim 9, wherein the voting for the first proposed value comprises changing a previous vote for the second proposed value if the second proposed value was previously voted for and if the second client identifier is less dominant than the first client identifier.

12) The computer-readable medium of claim 11, wherein the second proposed value comprises a second proposed function, and wherein the changing the previous vote comprises undoing a previous execution of the second proposed function.

13) The computer-readable medium of claim 11, wherein the second proposed value comprises a second proposed function, and wherein the changing the previous vote comprises allowing a previous provisional execution of the second proposed function to expire.

14) The computer-readable medium of claim 9, wherein the first proposed value comprises a first function identified by a first function identifier, and wherein the voting

for the first proposed value comprises executing the first function in the first system step unless the first function identifier is equivalent to a second function identifier that identifies a second function, wherein the second function was executed in a second system step that preceded the first system step.

15) The computer-readable medium of claim 9, wherein the first proposed value comprises a first idempotent function, and wherein the voting for the first proposed value comprises executing the first idempotent function in the first system step even if the first idempotent function is equivalent to a second idempotent function that was executed in a second system step that preceded the first system step.

16) The computer-readable medium of claim 9 having further computer-executable instructions for performing steps comprising: receiving a message, wherein the message is part of a fault tolerant consensus algorithm; ignoring additional proposed values from the first client; and participating in the fault tolerant consensus algorithm.

17) The computer-readable medium of claim 16, wherein the participating in the fault tolerant consensus algorithm comprises transmitting a possibly selected proposed value if a proposed value was previously voted for, wherein the possibly selected proposed value was previously voted for and was proposed by a client having a most dominant client identifier among all clients whose proposals were received and who proposed values for a current system step.

18) The computer-readable medium of claim 9 having further computer-executable instructions for performing steps comprising: transmitting one or more polling messages to initiate a fault tolerant consensus algorithm; receiving one or more vote indication messages in response to the one or more polling messages; and selecting, as a third proposed value, any value if the one or more vote indication messages indicate that at least one device has not previously voted or if the one or more vote indication messages indicate two or more different possibly selected proposed values, or selecting, as the third proposed value, a common possibly selected proposed value if all possibly selected proposed values indicated by the one or more vote indication messages are equivalent to the common possibly selected proposed value, wherein a possibly selected proposed value was previously voted for by a device and was proposed by a client having a most dominant client identifier among all clients whose proposals were received by the device, and wherein further the third proposed value is proposed using the fault tolerant consensus algorithm.

19) A computing device operating as part of a distributed computing system, the computing device comprising: a processing unit performing steps comprising: comparing a first client identifier to a second client identifier if a second proposed value, proposed by a second client having the second client identifier, was previously voted for in a first system step; and voting for a first proposed value in the first system step if the first client identifier is more dominant than the second client identifier and the second proposed value was previously voted for; and a network interface performing steps comprising: receiving the first proposed value from a first client having the first client identifier;

transmitting a first indication of the voting for the first proposed value to one or more devices also operating as part of the distributed computing system; and transmitting a first result of the voting for the first proposed value to the first client.

20) The computing device of claim 19, wherein the first proposed value comprises a first function, and wherein the voting for the first proposed value comprises provisionally executing the first function in the first system step.

21) The computing device of claim 19, wherein the voting for the first proposed value comprises changing a previous vote for the second proposed value if the second proposed value was previously voted for and if the second client identifier is less dominant than the first client identifier.

22) The computing device of claim 21, wherein the second proposed value comprises a second proposed function, and wherein the changing the previous vote comprises undoing a previous execution of the second proposed function.

23) The computing device of claim 21, wherein the second proposed value comprises a second proposed function, and wherein the changing the previous vote comprises allowing a previous provisional execution of the second proposed function to expire.

24) The computing device of claim 19, wherein the first proposed value comprises a first function identified by a first function identifier, and wherein the voting for the first

proposed value comprises executing the first function in the first system step unless the first function identifier is equivalent to a second function identifier that identifies a second function, wherein the second function was executed in a second system step that preceded the first system step.

25) The computing device of claim 19, wherein the first proposed value comprises a first idempotent function, and wherein the voting for the first proposed value comprises executing the first idempotent function in the first system step even if the first idempotent function is equivalent to a second idempotent function that was executed in a second system step that preceded the first system step.

26) The computing device of claim 19, wherein the processing unit performs further steps comprising: ignoring additional proposed values from the first client; and participating in a fault tolerant consensus algorithm; and wherein the network interface performs further steps comprising: receiving a message, wherein the message is part of the fault tolerant consensus algorithm.

27) The computing device of claim 26, wherein the participating in the fault tolerant consensus algorithm comprises transmitting a possibly selected proposed value if a proposed value was previously voted for, wherein the possibly selected proposed value was: previously voted for and was proposed by a client having a most dominant client identifier among all clients who proposed values to the computing device for a current system step.



28) The computing device of claim 19, wherein the processing unit performs further steps comprising: selecting, as a third proposed value, any value if one or more vote indication messages indicate that at least one device has not previously voted or if the one or more vote indication messages indicate two or more different possibly selected proposed values, or selecting, as the third proposed value, a common possibly selected proposed value if all possibly selected proposed values indicated by the one or more vote indication messages are equivalent to the common possibly selected proposed value, wherein a possibly selected proposed value was previously voted for by a device and was proposed by a client having a most dominant client identifier among all clients whose proposals were received by the device, and wherein further the third proposed value is proposed using the fault tolerant consensus algorithm; and wherein the network interface performs further steps comprising: transmitting one or more polling messages to initiate the fault tolerant consensus algorithm; and receiving the one or more vote indication messages in response to the one or more polling messages.

29) The computing device of claim 19, wherein the operating as part of the distributed computing system comprises operating as a client of the distributed computing system.

30) The computing device of claim 19, wherein the distributed computing system is comprised of devices that are also clients of the distributed computing system.

31) A conflict tolerant message delay reducing consensus algorithm for use in computing environment comprising a dedicated client device and a distributed computing system implemented by one or more devices, wherein the computing system implements the conflict tolerant message delay reducing consensus algorithm, the conflict tolerant message delay reducing consensus algorithm comprising: transmitting one or more proposed values from one or more clients; voting, at one or more of the one or more devices implementing the distributed computing system, for a proposed value from among the one or more proposed values, wherein the proposed value was proposed by a client having a most dominant client identifier from among the one or more clients proposing values; transmitting to one or more of the one or more devices implementing the distributed computing system an indication of the vote for the proposed value; and transmitting, to the client having the highest client identifier, a result of the vote for the proposed value.

32) The conflict tolerant message delay reducing consensus algorithm of claim 31, wherein the one or more devices implementing the distributed computing system also act as clients of the distributed computing system.

33) The conflict tolerant message delay reducing consensus algorithm of claim 31, wherein the dedicated client device is identified by a least dominant client identifier.

34) The conflict tolerant message delay reducing consensus algorithm of claim 31 further comprising: determining that the distributed computing system has selected the

proposed value when each of the one or more devices implementing the distributed computing system has voted for the proposed value.

35) The conflict tolerant message delay reducing consensus algorithm of claim 31, wherein the proposed value comprises a function, and wherein the voting for the proposed value comprises provisionally executing the function in a system step.

36) The conflict tolerant message delay reducing consensus algorithm of claim 31, wherein the voting for the proposed value comprises changing a previous vote if the previous vote was for a previously proposed value, proposed by a previous client having a client identifier that is less dominant than the client proposing the proposed value.

37) The conflict tolerant message delay reducing consensus algorithm of claim 31 further comprising: ending the conflict tolerant message delay reducing consensus algorithm and commencing a fault tolerant consensus algorithm if a failure is detected.

38) The conflict tolerant message delay reducing consensus algorithm of claim 37, wherein the failure is detected by a monitoring device, and wherein the computing environment further comprises the monitoring device.

39) The conflict tolerant message delay reducing consensus algorithm of claim 37, wherein the commencing the fault tolerant consensus algorithm comprises identifying a possibly selected proposed value, wherein the possibly selected proposed value is any

value if at least one of the one or more devices implementing the distributed computing system did not previously vote or, if at least one of the one or more devices implementing the distributed computing system previously voted, then the possibly selected proposed value was previously voted for by at least one of the one or more devices implementing the distributed computing system and was proposed by the client having the most dominant client identifier from among the one or more clients proposing values, for a current system step, to the at least one of the one or more devices implementing the distributed computing system.